3D Data Acquisition and Registration using Two Opposing Kinects

Vahid Soleimani, Majid Mirmehdi, Dima Damen, Sion Hannuna, Massimo Camplani
Visual Information Laboratory
Faculty of Engineering, University of Bristol, BS8 1UB, UK
Vahid.Soleimani@bristol.ac.uk

Abstract

We present an automatic, open source data acquisition and calibration approach using two opposing RGBD sensors (Kinect V2) and demonstrate its efficacy for dynamic object reconstruction in the context of monitoring for remote lung function assessment. First, the relative pose of the two RGBD sensors is estimated through a calibration stage and rigid transformation parameters are computed. These are then used to align and register point clouds obtained from the sensors at frame level. We validated the proposed system by performing experiments on known-size box objects with the results demonstrating accurate measurements. We also report on dynamic object reconstruction by way of human subjects undergoing respiratory functional assessment.

1. Introduction

Recent affordable RGBD sensors have provided opportunities and inroads in various areas of computer vision, including 3D object and scene reconstruction. Methods for capturing the full extent of an object, or a complete scene, have been proposed using handheld sensors and temporal fusion [22, 29, 49]. Alternatively, static multi-sensor setups with varying overlapping requirements between the sensors have been proposed to reconstruct dynamic scenes on frame-level basis [7,16,24,28,35,41]. These avoid the need, and challenge, for alignment and fusion between frames and can readily reconstruct dynamic scenes and deformable objects in real-time. For example, Kowalski *et al*. [24] recently presented a 3D data acquisition system, using up to four Kinect V2 sensors, in which they manually calibrated their system to register the point clouds from each sensor in a two-step procedure involving rough estimation and refinement. Their qualitative-only results showed good performance for general static and dynamic object reconstruction. However, their calibration stage is cumbersome requiring self-designed markers, manual labelling of marker's locations, and sufficient overlap between the sensors.

While this work belongs to the category of a static multi-sensor setup, we rely on a simplified approach of using two static *opposing* RGBD sensors with negligible overlap. Each sensor can perceive *nearly* half of the object, resulting in frame-level reconstruction of dynamic objects. This is valuable in applications where a less intrusive and more easily configured setup is necessary, which inherently means as few sensors as possible and as little inconvenience to the subject as possible. One such application is within healthcare pertaining to respiratory measurements or pulmonary function testing, for which depth-based approaches have recently emerged [31, 32, 44, 47], albeit for a single sensor.

The proposed approach is able to reconstruct rigid and dynamic objects to high accuracy, which we evaluate quantitatively on rigid objects and qualitatively on animated subjects. Ease of setup and high accuracy (range of average errors is $0.21 - 0.84$ cm across 3 objects and 3 placements) is achieved through, (a) a fast and automatic calibration process using double-sided calibration chessboards placed at varying depths, (b) synchronising intra-Kinect RGB and depth channels as well as two data acquisition machines, and (c) a highly accurate point cloud registration approach using only the infrared stream to specify real world coordinates, as opposed to using RGB and depth which is likely to increase registration error.

The main contributions of our work are twofold. First, the deployment of only two Kinect sensors for 3D data capture minimises the overall operation space, reduces the system setup and calibration effort, lowers system costs, and minimises the temporal frame alignment error. Second, unlike many other previous approaches, which require a considerable overlap between point clouds for registration [23, 24, 28, 35, 41], our proposed method is able to perform temporal and spatial alignment of two non-overlapping point clouds. Our proposed method is open source[1].

The rest of this paper is organised as follows. After considering some previous works in Section 2, the proposed 3D data acquisition system is explained in Section 3. Exper-

[1] https://github.com/BristolVisualPFT/

iments, results and discussion are presented in Section 4, before concluding the paper in Section 5.

2. Related works

There are many existing works on the registration of multiview range images obtained by photometric stereo and structured light techniques, such as [4, 17, 37, 39, 46], with some summative works *e.g*. in [18, 42, 43]. We limit this review to methods using affordable commercial commodity RGBD sensors, such as the Kinect, for multiview 3D reconstruction and registration using single and multiple RGBD sensors.

Single RGBD sensor 3D reconstruction – Approaches which apply a single capturing device, either use a moving sensor on a path around the object or the object rotates for a fixed position sensor. These approaches apply point matching algorithms, mainly Iterative Closest Point (ICP) [9, 48] and other adapted variants [34], to register point clouds by minimising the distance between continuously detected corresponding keypoints in consecutive keyframes. These corresponding keypoints can be determined using uniform sampling of point clouds, general 2D features *e.g*. Scale Invariant Feature Transform (SIFT [26]) and Speeded Up Robust Features (SURF [6]), or depth features specifically designed for 3D registration *e.g*. Fast Point Feature Histograms [38].

Some approaches [21, 22, 29, 30, 49] have recently been proposed for reconstruction of non-rigid objects and scenes using a single RGBD sensor. Izadi *et al*. [22] and Newcombe *et al*. [30] introduced KinectFusion as a real-time 3D reconstruction approach using a moving Kinect. They presented a new GPU pipeline which allows for real-time camera tracking, surface reconstruction, and rendering. However, These methods expect a static scene during reconstruction. In [49], Zollhöfer *et al*. first acquired an initial template using KinectFusion of [22], for which the object needed to be static for $\sim$1 minute. Next, in the non-rigid reconstruction phase, for each frame they roughly aligned the template to the input data and then fitted the non-rigid surface using a new efficient GPU-based Gauss-Newton solver, which minimised the fitting energy function. Newcombe *et al*. [29] presented a real-time DynamicFusion technique for tracking surfaces and dynamic reconstruction of non-rigid objects. Each live depth frame was fused into a canonical space using an estimated volumetric warp field, which removed the scene motion, and a truncated signed distance function volume reconstruction was obtained. However, since they omitted the RGB stream and also not utilized global features, their method fails to track surfaces in specific types of topological changes (*e.g*. closed to open hands) and it is also prone to drift. In the most recent work we are aware of, Innmann *et al*. [21] proposed a similar method to [29], in which they tried to address these issues. In addition to the dense depth correspondences, they applied global sparse color-based SIFT feature correspondences which allows them to better deal with drifts and improve tracking.

Although these single RGBD sensor approaches yield highly impressive results, they are not able to capture changes that simultaneously happen in those parts of the object that are not within the field of view. Further, they require that there should be a substantial overlap in the depth data of consecutive frames, which enables the point matching algorithm, *e.g*. ICP, to better estimate the point cloud registration parameters. Finally, they can be restricted in type and speed of deformations, *e.g*. fast and large deformations.

Multiple RGBD sensor 3D reconstruction – In these approaches, multiple static RGBD sensors are co-located to simultaneously capture the scene from different points of view. To be able to find the sensors' relative pose, they need to be calibrated individually and together using optical and/or geometric techniques. In the former, calibration patterns or markers, like a chessboard, are typically observed by the cameras. In the latter, rigid transformations that help align the 3D point clouds from each RGBD camera are computed.

Both optical and geometric techniques were used in [23] to register point clouds from multiple Kinects. First, the relative pose of the Kinects was approximated using a customised calibration box with 2D visual markers attached on each side. After the point clouds were roughly aligned, they applied an adapted version of KinectFusion [22] in an extra refinement step to create the final point cloud. Similar to Kowalski *et al*. [24], relative position of markers had to be computed manually.

Miller *et al*. [28] suggested an unsupervised method to estimate the rigid transform parameters of two overlapping RGBD sensors, without any initial calibration. First, a moving foreground object was detected in the scene captured by both sensors and the point clouds were roughly aligned by using the centroid of moving foreground objects. In a refinement stage, they tuned the estimation by optimising an energy function which used a nearest-neighbour penalty across all frames. However, this penalty had negative effects where there was not sufficient overlap between the point clouds. To deal with this problem, they added a free space violation term to the nearest-neighbour analysis.

Deng *et al*. [16] localized rigid transformation parameters to improve registration accuracy of point clouds obtained from two Kinects. A 3D grid of translation and rotation parameters was first constructed using the established correspondence points obtained from a moving chessboard, and then interpolated. The authors reported improvements in their point cloud registration accuracy in comparison to global rigid transformation approaches. However, their

method demands that there should be a huge amount of overlap in the data captured by the two sensors, and their local registration results in geometrical distortion in the final reconstructed point cloud.

Avetisyan *et al*. [5] employed an optical tracking system to increase depth measurement accuracy of three inward and circularly-located RGBD sensors. A tracked chessboard was moved through the capture space in front of each sensor and a separate lookup table was created for each sensor. This lookup table consisted of the chessboard crossing points locations in the tracker coordinate system and their corresponding locations in the sensor coordinate system. The lookup table was then used to correct the sensors' depth measurement accuracy during scene reconstruction. The depth sensors and the optical tracking system were still calibrated using a single rigid transformation.

In [41], Seifi *et al*. presented a geometric registration approach, which exploits image-based features to align overlapping point clouds obtained from two RGBD sensors capturing the scene. Matching keypoints were detected from corresponding RGB images of both sensors using SURF and ORB [36] feature descriptors. These keypoints were then refined to reject incorrectly matched keypoints. Finally, after finding the corresponding location of these matched keypoints in depth space, rigid transformation parameters were estimated. However, like all geometric approaches, such as [23, 28, 35, 41], there is a dependency on the availability of good features and a considerable amount of overlap in the sensors' capture space.

Most recently, Beck and Froehlich [7] proposed a volumetric method to calibrate multiple RGBD sensors by transforming each depth sensor space into a normalized volume space, performing a reference sampling and interpolation. A chessboard was placed in various locations of the capturing volume and captured by RGBD sensors while simultaneously being tracked by a motion capture system. Real world location of chessboard crossing points in both RGBD sensors and motion capture system were used to fill a 3D lookup table with a typical size of $128\times128\times256$. After performing an interpolation to fill empty cells, the table was used in the reconstruction stage. Approximately 2000 reference samples were required for a capturing volume of about $1.5m\times1.8m\times1.5m$, which takes 20-30 minutes to be performed.

In our proposed optical data acquisition and registration approach, we perform a 1-step, fast and accurate calibration (with no refinement step needed) by using three double-sided chessboards at different depths in the scene and only a pair of infrared/depth images taken by each Kinect.

3. Proposed method

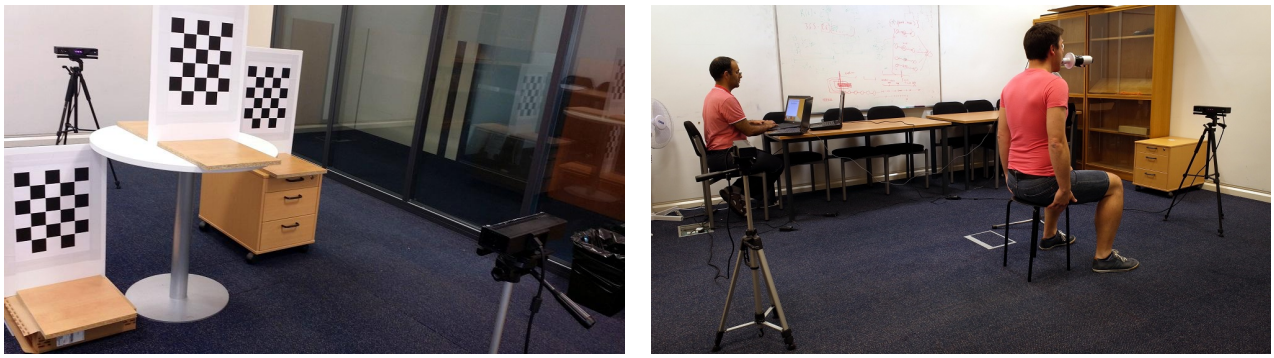
We propose a 3D data acquisition and registration system that uses two opposing (i.e. facing) Kinects. We establish the Kinect's optimal measuring distance to minimise signal noise (as outlined in Section 4.1). In the calibration stage, the crossing points of three double sided chessboards, placed at different depths from the two facing Kinects, are detected automatically. Then, in the registration stage, rigid transformation parameters are computed, which are used to transfer the two Kinects' point clouds to a joint coordinate system in the registration and reconstruction stage.

3.1. Calibration stage

System configuration and setup – We used two Kinects facing each other with $\sim3m$ distance between them (Fig. 1), allowing objects to be captured at the optimal distance away from each sensor. For registering and aligning two sets of 3D points, we need at least three corresponding and distinct 3D points in each point set [9]. Using more distant points, which are not at the same depth from the sensor, makes alignment more accurate and decreases registration error. Thus, to help with the calibration, we used three double-sided chessboards which were placed at different depths from the Kinects (Fig. 1). To make a double sided chessboard, a 5×6 pattern (with chessboard square size of $55 \times 55mm^2$) was printed on two $A3$ papers, which were then joined back to back and held by a frame such that the chessboards' crossing points were aligned as precisely as possible. This solution provides us with three groups of points (3×20 inner points in total) so that the points in any group have different (x, y, z) coordinates from points in the other groups.

Data acquisition and synchronization – Unlike Kowalski *et al*. [24], our system was designed to capture all four of RGB, depth, infrared and body joints data in simultaneous processing threads at full frame rate (30*fps*). Online visualisation is possible, although at the expense of lower frame rate. Our proposed system is able to generate RGB point clouds from pre-recorded and synchronised RGB and depth data.

Here, we wish to achieve 'synchronization' between corresponding frames of different data modalities in each Kinect separately (intra-Kinect) and also, between corresponding frames of the same type in different Kinects (inter-Kinect). Intra-Kinect synchronization is necessary to identify temporally corresponding RGB, depth, and skeleton data frames in each Kinect, which was simply performed by using the timestamps provided for each data frame in a Kinect. Inter-Kinect synchronization was achieved by synchronising the system time of two locally networked PCs (one for each Kinect) using Network Time Protocol (NTP) and recording the Kinects' system and threads timestamps for aligning each data frame. Since there are no means of triggering multiple Kinects simultaneously by software control commands, this can cause a maximum lag of $30ms$ between our two Kinects, which would cause a synchro-



(a) (b)

Figure 1: (a) Two Kinects and three chessboards setup, (b) Applying the proposed system to capture and reconstruct a subject performing pulmonary function testing using a spirometer.

nization error of at most one frame. We reduced this error by sending only one trigger command at the beginning of the capture, from one machine to another through the network, however the error is dependant on the network traffic and speed. Note that, the more Kinect RGBD sensors within a system, as in [1–3, 7, 8, 24, 27], the greater is this error.

Lens distortion correction – Kinect V2 depth images are computed from the captured infrared images and therefore, both images have the same optical specifications. Similar to other lens-based imaging devices, the Kinect also suffers from lens distortion. Thus, both the infrared and depth image distortions were corrected by applying the Brown model [12].

Establishing crossing points correspondences – We used real-world coordinates of the crossing points of three double sided chessboards (3×20 points) to align point clouds and register them to a joint coordinate system. Previous approaches [1–3, 23, 24, 27] have used both RGB and depth sensor data to obtain real world coordinate of points required for calibration. However, we detect the real world coordinates of the crossing points [19] from intensity and depth space obtained by illumination-normalised infrared images and depth images, respectively. Using only infrared sensor instead of using both RGB and depth, increases point cloud registration accuracy by eliminating the error caused by RGB to depth space mapping. Fig. 2 shows the detected chessboards' crossing points where the corresponding crossing point sets in the two Kinects' infrared images are indicated in the same color.

Kinects pose estimation – As there is an insufficient number of overlapping points in our point clouds, an iterative point matching algorithm, like ICP [9, 48], is unsuitable for aligning them. Thus, we considered one Kinect's coordinate system as reference, and then the other Kinect's relative pose was estimated using translation $\mathbf{T}$ and rotation $\mathbf{R}$ transformations. $\mathbf{T}$ and $\mathbf{R}$ were computed by registering 20 corresponding crossing points of the reference and the second Kinect, *i.e.* $\mathbf{Q}$ and $\mathbf{Q}'$, as

$$\mathbf{Q}' = \mathbf{R} \times \mathbf{Q} + \mathbf{T}. \tag{1}$$

The rotation matrix $\mathbf{R}$ is computed by applying singular value decomposition on a cross-covariance matrix $\mathbf{M}$ created using $\mathbf{Q}$ and $\mathbf{Q}'$ point sets [13],

$$\mathbf{M} = \frac{1}{N} \sum_{j=1}^{N} \left[(\mathbf{Q}_j - \mathbf{Q}_\mu)(\mathbf{Q'}_j - \mathbf{Q'}_\mu)^T \right], \tag{2}$$

where $\mathbf{Q}_j$ and $\mathbf{Q'}_j$ denote the jth points in $\mathbf{Q}$ and $\mathbf{Q}'$ point sets, $\mathbf{Q}_\mu$ and $\mathbf{Q'}_\mu$ are the point sets' centroids, and $N = 60$ is the number of points in each set. Since $\mathbf{M}$ is a real square matrix with a positive determinant, it can be decomposed into orthogonal square matrices $\mathbf{U}$ and $\mathbf{V}$, and diagonal non-negative matrix $\mathbf{\Sigma}$, such that $\mathbf{M} = \mathbf{U}\mathbf{\Sigma}\mathbf{V}^T$ [20], where $\mathbf{\Sigma}$, and $\mathbf{U}$ and $\mathbf{V}^T$, are considered as scaling matrix and rotation matrices, respectively. Thus, $\mathbf{M}$ can be intuitively interpreted as a geometrical transformation composed of a rotation, a scaling, and another rotation. As the transformations need to be rigid, we omit $\mathbf{\Sigma}$ to preserve the objects' shape and size. Thus, the rotation matrix is $\mathbf{R} = \mathbf{U}\mathbf{V}^T$ and the translation matrix is $\mathbf{T} = -\mathbf{R} \times \mathbf{Q}_\mu + \mathbf{Q'}_\mu$.

Fig. 3 shows the aligned scenes from each Kinect.

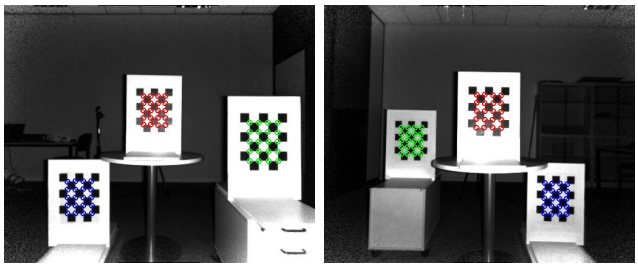


Figure 2: Establishing crossing points correspondences.

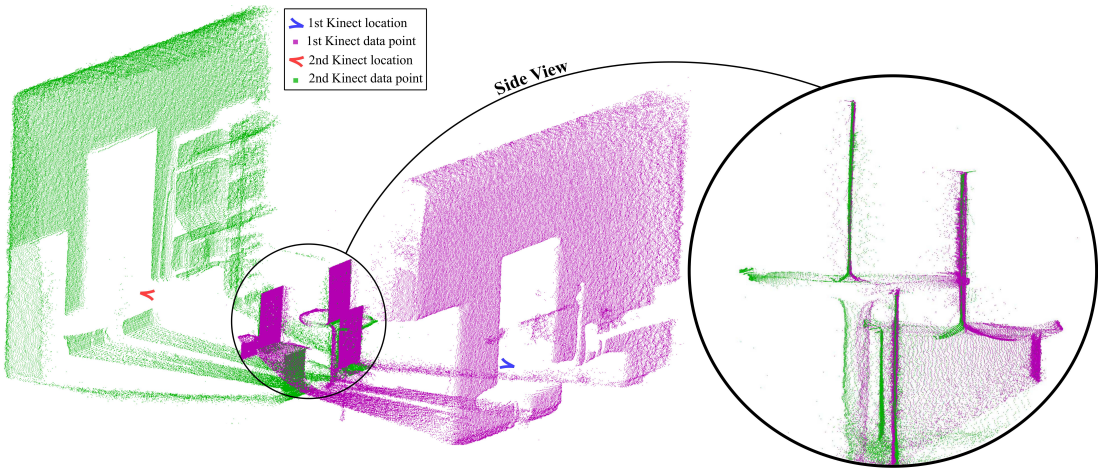


Figure 3: Facing Kinects: the scene as viewed by each Kinect, after alignment. The side view shows the points on the calibration chessboards.

3.2. Registration and reconstruction

For a pair of temporal sequences of our dynamic object captured by our Kinects, *e.g.* of a human being breathing forcefully through a spirometer, we first found the corresponding frames using our intra-Kinect and inter-Kinect synchronisation approach. Then, the reference point cloud $\mathbf{P}_1^{ref}$ and the second point cloud $\mathbf{P}_2$, were generated. Using the computed translation and rotation matrices, we transformed $\mathbf{P}_2$ into the coordinate system of $\mathbf{P}_1^{ref}$, such that $\mathbf{P}_2' = \mathbf{R} \times \mathbf{P}_2 + \mathbf{T}$. Finally, we created a merged point cloud, $\mathbf{P}$, as our proposed reconstructed point cloud:

$$\mathbf{P} = \mathbf{P}_1^{ref} \bigcup \mathbf{P}_2'. \tag{3}$$

4. Experimental results and discussion

Two laptops with Intel® Core™ i7 quad core processors running at 2.8GHz and 16GB memory were used to acquire the data streams from our two Kinects. The proposed approach, comprising data acquisition, registration, reconstruction, and visualization were implemented in Microsoft Visual Studio 2012, using OpenCV [10] and Visualization Toolkit [40] libraries and Matlab 2015b. For concurrent processing, we used the Intel® Threading Building Blocks library [33] to grab, buffer and record RGB and depth, and body joint data in separate threads which enabled us to reconstruct a 3D dynamic object at a consistent 30*fps*.

4.1. Noise analysis

Kinect depth estimation suffers from measurement noise caused by the depth sensor technology. Since the Kinect V2 was released only recently, there is little public information on the nature and characteristics of its noise. We performed a planar noise analysis to find the optimal distance range between the sensor and the subject.

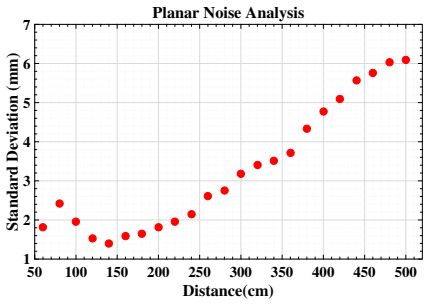


Figure 4: Planar surface noise analysis with distance range of $60 - 500cm$.

In this experiment, we estimated the sensor measurement error by placing the Kinect at various distances - from $60cm$ to $500cm$ at $20cm$ intervals - in front of a white wall under normal room temperature and lighting conditions, with the sensors optical axis approximately perpendicular to the wall. At each position, a sequence of 200 frames were recorded and $15K$ depth values were randomly sampled from a constant-size patch at the center of the sensor's viewpoint and the standard deviation was computed for them. Figure 4 illustrates this standard deviation in mm plotted against the sensor distance to the wall. It shows a non-linear behaviour similar to the general ToF depth sensors [25]. Furthermore, a similar noise curve has been reported for the same sensor by Breuer *et al.* [11]. Noise increases between 60 and $80cm$, then drops to its minimum at $\sim 150cm$. Accordingly, we carried out our experiments with the Kinect placed within a range of $\sim 150 - 200cm$ from the object depending on the size of the object.

4.2. Assessing measurement accuracy

Figure 3 shows the point clouds, in purple and green, obtained by our two Kinects after registration. An en-

larged side-view of the three calibration chessboards, held by 5mm-thick frames, is also shown, after matching and alignment. We assessed the accuracy of our calibration method by computing the root mean square error (RMSE) of distances between corresponding chessboards' crossing points in $\mathbf{P}_1^{ref}$ and $\mathbf{P}_2'$. This error was computed as $4.6mm$ for the lung function test setup ($\sim 3m$ distance between Kinects at a height of $\sim 0.6m$).

We also evaluated calibration accuracy by estimating the dimensions of three boxes of known sizes placed at different depths and validating them against the groundtruth values. Since we needed more capturing space to be able to position the boxes in the scene, in this experiment the Kinects were placed at $\sim 4m$ away from each other and at a height of $\sim 1.2m$. The RMSE of distances between corresponding crossing points in this setup was computed as $6.8mm$.

We evaluate spatial registration accuracy of the proposed method by measuring dimensions and volume, and performing surface analysis, of the three differently sized boxes (Figure 5). Each box was captured three times, i.e. once at each of three different depths or locations, and all measurements made. Table 1 presents the three locations at which (the centroid of) each box was placed in the world coordinate system and the real dimensions of the three boxes. In each of the nine captured sequences, the box was segmented from the registered point clouds by depth value thresholding. For each reconstructed box, sides planarity and orthogonality, height, width, depth, and volume were automatically estimated by performing surface analysis, and then compared against groundtruth measurements.

The boxes' four sides were automatically apportioned into separate point sets using the M-estimator SAmple Consensus (MSAC) approach [45]. Then, a plane was fitted on the point set of each side (see Fig. 6) using a first degree polynomial, and R-squared and RMSE were computed for the fitted plane. The angles between the sides were esti-

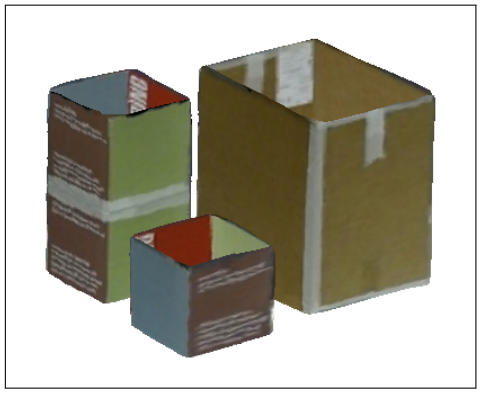


Figure 5: 3D reconstruction of the boxes used to evaluate the proposed system accuracy.

	X	Y	Z
Location 1	-17.7	18.7	241
Location 2	-41.4	23.7	202
Location 3	10.0	41.1	166

	W	H	D
Box1	34.0	47.0	43.5
Box2	23.2	45.0	23.2
Box3	23.2	22.5	23.2

Table 1: Centroid location of the 3 boxes in the Kinects' joint coordinate system and their actual dimensions (in *cm*).

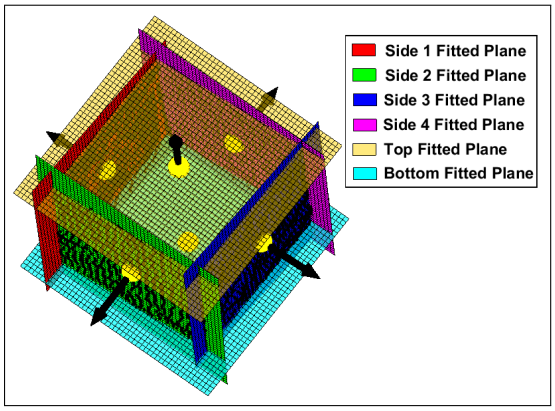


Figure 6: Plane fitting of *Box3* sides.

mated using the normal vectors of the fitted planes. Tables 2, 3, and 4 present these estimations for *Box1*, *Box2* and *Box3* in the three locations (S1-4 refer to 4 sides of each box). The R-squared and RMSE values illustrate that the side-planarity is preserved very well in the reconstructed models. Furthermore, the estimated angles between the fitted planes show that our proposed system performs well in measuring orthogonality.

To help achieve the best estimation of height, width, and depth automatically, we used the planes fitted on the lateral sides of the boxes. Since the bottom of the boxes was not captured and the top was too sparse to rely on, the corresponding planes were computed by inference from the existing sides. First, the cross product of the normals of the lateral sides' fitted planes were computed to define two planes perpendicular to the side planes. These planes, which represent the bottom and top planes, were then placed respectively at the bottommost and topmost of the box's point cloud, where there is a significant change in the number of points. Then, the eight corner points of the boxes were computed using the intersection of the fitted planes. Since the fitted planes are not exactly parallel, height, width, and depth were estimated by computing the average distance between the relevant four corner points of each side-plane and its facing side-plane.

Even though the box volume can be approximated using the estimated height, width, and depth ($V = W \times H \times D$), we estimated the volume by applying Gauss's Divergence Theorem as described in [44], since that would have to be used for geometrically non-uniform or non-rigid objects in any

	Location 1		Location 2		Location 3	
	R-Squared	RMSE	R-Squared	RMSE	R-Squared	RMSE
S1	0.997	0.003	0.997	0.004	0.997	0.003
S2	0.998	0.004	0.998	0.004	0.998	0.003
S3	0.996	0.004	0.997	0.003	0.997	0.003
S4	0.998	0.004	0.998	0.003	0.997	0.004
	Estimated Angle		Estimated Angle		Estimated Angle	
S1–S2	88.89°		89.45°		89.17°	
S1–S4	89.41°		88.52°		88.42°	
S2–S3	89.19°		89.16°		89.94°	
S3–S4	89.10°		88.18°		90.83°	
S1–S3	0.59°		1.25°		1.08°	
S2–S4	2.35°		2.56°		1.90°	

Table 2: Surface analysis of *Box1* using plane fitting

	Location 1		Location 2		Location 3	
	R-Squared	RMSE	R-Squared	RMSE	R-Squared	RMSE
S1	0.994	0.004	0.997	0.002	0.993	0.004
S2	0.997	0.002	0.992	0.004	0.995	0.003
S3	0.995	0.003	0.993	0.005	0.993	0.004
S4	0.992	0.005	0.996	0.003	0.996	0.003
	Estimated Angle		Estimated Angle		Estimated Angle	
S1–S2	88.85°		89.96°		88.78°	
S1–S4	88.89°		89.02°		89.27°	
S2–S3	89.88°		89.26°		88.82°	
S3–S4	89.92°		89.81°		89.32°	
S1–S3	1.09°		0.78°		0.46°	
S2–S4	0.24°		1.37°		1.05°	

Table 3: Surface analysis of *Box2* using plane fitting

	Location 1		Location 2		Location 3	
	R-Squared	RMSE	R-Squared	RMSE	R-Squared	RMSE
S1	0.997	0.003	0.998	0.002	0.996	0.003
S2	0.990	0.004	0.990	0.004	0.995	0.004
S3	0.995	0.003	0.996	0.003	0.997	0.002
S4	0.992	0.004	0.991	0.004	0.994	0.004
	Estimated Angle		Estimated Angle		Estimated Angle	
S1–S2	88.65°		88.93°		89.19°	
S1–S4	88.45°		88.40°		89.22°	
S2–S3	88.99°		88.88°		88.64°	
S3–S4	91.20°		91.64°		88.67°	
S1–S3	0.51°		0.18°		0.71°	
S2–S4	0.37°		0.63°		0.54°	

Table 4: Surface analysis of *Box3* using plane fitting

case. To be able to perform the surface integral over the box boundary, the box surface was reconstructed by applying a 2D Delaunay triangulation [14] on the registered point cloud. Note that dimensions and volume are presented in centimetres and litres, respectively.

		Location 1		Location 2		Location 3	
		Estimated	Error	Estimated	Error	Estimated	Error
Box1	**W**	34.29	0.29	34.24	0.24	34.33	0.33
	H	47.18	0.18	47.16	0.16	47.27	0.27
	D	44.44	0.94	44.36	0.86	44.20	0.70
	V	69.75L	0.24L	69.31L	0.20L	68.28L	0.23L
Box2	**W**	23.98	0.78	23.75	0.55	23.85	0.65
	H	45.29	0.29	44.89	0.11	45.38	0.38
	D	24.02	0.82	23.91	0.71	24.14	0.94
	V	24.74L	0.52L	24.91L	0.69L	24.93L	0.71L
Box3	**W**	23.88	0.68	23.85	0.65	23.91	0.71
	H	22.68	0.09	22.38	0.11	22.69	0.29
	D	24.16	0.96	24.14	0.94	23.89	0.69
	V	12.65L	0.54L	12.51L	0.40L	12.48L	0.37L

Table 5: Automatically estimated width, height, depth (in *cm*) and volume (in *Litre*) of boxes using surface analysis.

Table 5 reports the estimated dimensions, volume, and their L_2 error for *Box1*, *Box2* and *Box3* against the groundtruth at each of the three locations. We note that the extent of the error is a little different across each dimension with an average L_2 error for the three boxes in all locations across height at 0.21, width at 0.54, and depth at 0.84. Considering there is a $\sim 4m$ distance between the two Kinects, our results show very good accuracy for the estimated measurements, independent of the location of the boxes.

4.3. Dynamic object reconstruction

We also demonstrate the ability of the proposed method to achieve dynamic 3D object reconstruction via two different examples. The first is based on dynamic human trunk 3D reconstruction for use in remote respiratory monitoring system. A relatively new area in remote depth-based lung function assessment using a single RGBD sensor is in formation, exemplified by [15, 31, 32, 44, 47]. These methods attempt to simulate traditional breathing tests, such as spirometry, however, none of these methods is able to decouple the subject's trunk motion from the subject's chest surface motion, which greatly affects the test results. Acquiring accurate and dynamic 3D body shape using our proposed method during the breathing test, can better address this problem. In this test, the distance between each Kinect and the subject was $\sim 1.5m$ (optimal distance), at a height of $\sim 0.6m$, to be able to observe chest motion as accurately as possible. Then, a 3D surface of the subject's trunk performing a real lung function assessment test, *i.e.* Forced Vital Capacity (FVC), was reconstructed per frame. The analysis of such data demands precise point cloud alignment, accurate temporal frame synchronization, body joints data acquisition to estimate body pose, and consistent full frame rate

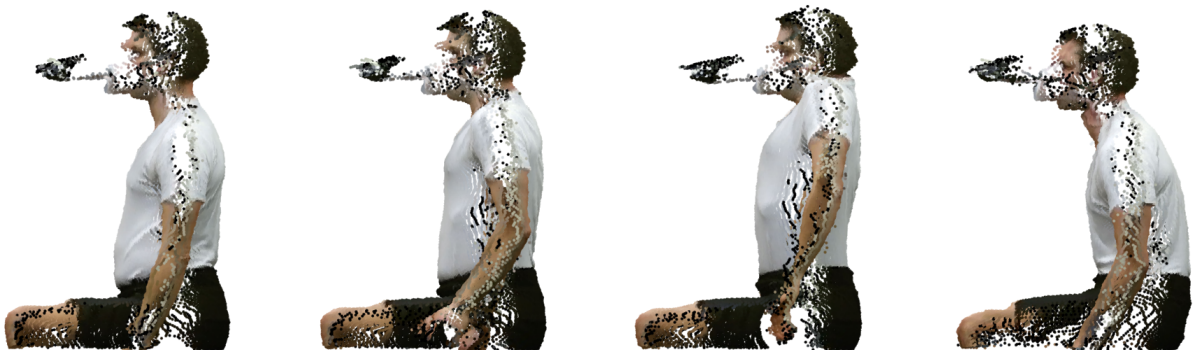


Figure 7: Dynamic 3D reconstruction of a subject's trunk performing lung function test using a spirometer.

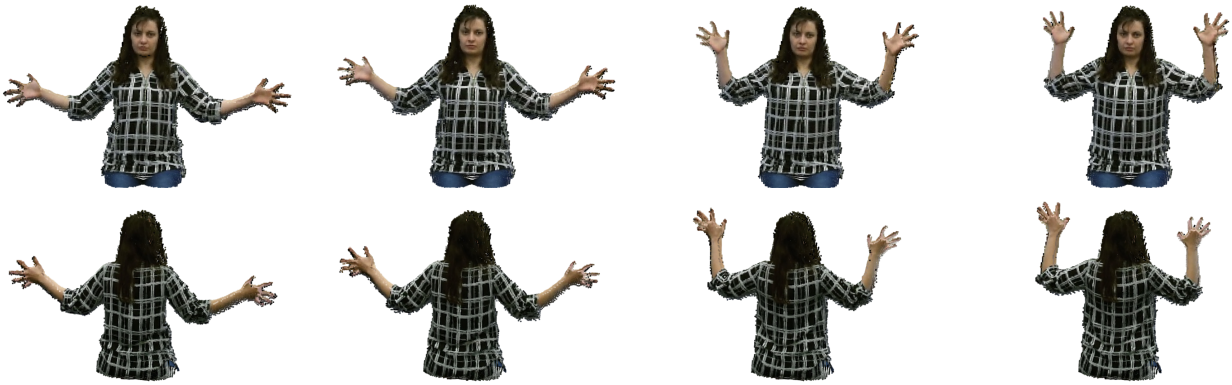


Figure 8: Dynamic 3D reconstruction of a subject waving hands.

(30*fps*) recording, all of which are provided by our system. Fig. 7 shows sample 3D reconstructed frames of a subject performing the FVC test. The reconstructions enable monitoring of the subject's trunk during the test. Although the gap between the two aligned point clouds is not important in this application, it can be filled by interpolation.

The second example was performed to show accurate temporal and spatial point cloud alignment by way of the subject performing different actions, *e.g*. waving hands, dancing, and jumping. The two facing Kinects were placed $\sim 4m$ away from each other at a height of $\sim 1.2m$. Sample 3D reconstruction of a subject waving hands in different frames are presented in Fig 8. As can be seen, the fingers have been well aligned and reconstructed.

5. Conclusion

We proposed a 3D RGBD data acquisition system which can provide accurate temporal and spatial 3D reconstruction that can be used in applications such as remote respiratory monitoring and lung function assessment. The extrinsic parameters of the two facing Kinects were computed in a calibration stage, using three double-sided chessboards placed at varying depths. Then, these parameters were exploited to register point clouds and reconstruct 3D, dynamic objects, for example performing lung function testing using a spirometer, and other actions such as waving. We evaluated the proposed system's accuracy by automatically measuring the dimensions, volume, and surface information of three different boxes and showed that it is efficient in reconstructing the boxes and estimating their dimensions. Compared to the currently existing state-of-the-art dynamic 3D data acquisition approaches, our proposed system only uses two sensors achieving frame-level reconstruction suitable for capturing fast and abrupt motions of dynamic objects. One shortcoming of our approach is that the current arrangement of our Kinects can result in missing information on parts of the object obscured from the Kinects' view (*e.g*. see the side of the person's trunk in Fig. 7). However, The system has been designed such that it can be easily extended by more Kinects as long as each Kinect can see the three chessboards.

Acknowledgment

The authors would like to thank the University of Bristol Alumni Foundation for funding this research.

References

[1] H. Afzal, D. Aouada, D. Font, B. Mirbach, and B. Ottersten. Rgb-d multi-view system calibration for full 3d scene reconstruction. In *International Conference on Pattern Recognition*, pages 2459–2464, Aug 2014.

[2] D. Alexiadis, D. Zarpalas, and P. Daras. Fast and smooth 3d reconstruction using multiple rgb-depth sensors. In *IEEE Conference on Visual Communications and Image Processing Conference*, pages 173–176, Dec 2014.

[3] D. S. Alexiadis, D. Zarpalas, and P. Daras. Real-time, full 3-d reconstruction of moving foreground objects from multiple consumer depth cameras. *IEEE Transactions on Multimedia*, 15(2):339–358, Feb 2013.

[4] D. G. Aliaga and Y. Xu. A self-calibrating method for photogeometric acquisition of 3d objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 32(4):747–754, April 2010.

[5] R. Avetisyan, M. Willert, S. Ohl, and O. Staadt. Calibration of depth camera arrays. In *Proceedings of SIGRAD, Visual Computing*, number 106, pages 41–48, 2014.

[6] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. Speeded-up robust features (surf). *Computer vision and image understanding*, 110(3):346–359, 2008.

[7] S. Beck and B. Froehlich. Volumetric calibration and registration of multiple rgbd-sensors into a joint coordinate system. In *IEEE Symposium on 3D User Interfaces*, pages 89–96, March 2015.

[8] S. Beck, A. Kunert, A. Kulik, and B. Froehlich. Immersive group-to-group telepresence. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):616–625, April 2013.

[9] P. J. Besl and H. D. McKay. A method for registration of 3-d shapes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):239–256, Feb 1992.

[10] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[11] T. Breuer, C. Bodensteiner, and M. Arens. Low-cost commodity depth sensor comparison and accuracy analysis. In *SPIE Security+ Defence*. International Society for Optics and Photonics, 2014.

[12] D. C. Brown. Close-range camera calibration. *Photogrammetric Engineering*, 37(8):855–866, 1971.

[13] J. H. Challis. A procedure for determining rigid body transformation parameters. *Journal of biomechanics*, 28(6):733–737, 1995.

[14] M. de Berg, O. Cheong, M. van Kreveld, and M. Overmars. *Computational Geometry: Algorithms and Applications*. Springer-Verlag, 3rd edition, 2008.

[15] W. de Boer, J. Lasenby, J. Cameron, R. Wareham, S. Ahmad, C. Roach, W. Hills, and R. Iles. Slp: A zero-contact non-invasive method for pulmonary function testing. In *Proceedings of the British Machine Vision Conference*, pages 85.1–85.12, 2010.

[16] T. Deng, J. C. Bazin, T. Martin, C. Kuster, J. Cai, T. Popa, and M. Gross. Registration of multiple rgbd cameras via local rigid transformations. In *IEEE International Conference on Multimedia and Expo*, pages 1–6, July 2014.

[17] C. H. Esteban, G. Vogiatzis, and R. Cipolla. Multiview photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(3):548–554, March 2008.

[18] Y. Furukawa and C. Hernndez. Multi-view stereo: A tutorial. *Foundations and Trends in Computer Graphics and Vision*, 9(1-2):1–148, 2015.

[19] A. Geiger, F. Moosmann, . Car, and B. Schuster. Automatic camera and range sensor calibration using a single shot. In *IEEE International Conference on Robotics and Automation*, pages 3936–3943, May 2012.

[20] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, USA, 1996.

[21] M. Innmann, M. Zollhöfer, M. Nießner, C. Theobalt, and M. Stamminger. Volumedeform: Real-time volumetric non-rigid reconstruction. *arXiv preprint arXiv:1603.08161*, 2016.

[22] S. Izadi, D. Kim, O. Hilliges, D. Molyneaux, R. Newcombe, P. Kohli, J. Shotton, S. Hodges, D. Freeman, A. Davison, and A. Fitzgibbon. Kinectfusion: Real-time 3d reconstruction and interaction using a moving depth camera. In *Proceedings of the 24th Annual ACM Symposium on User Interface Software and Technology*, pages 559–568, 2011.

[23] B. Kainz, S. Hauswiesner, G. Reitmayr, M. Steinberger, R. Grasset, L. Gruber, E. Veas, D. Kalkofen, H. Seichter, and D. Schmalstieg. Omnikinect: Real-time dense volumetric data acquisition and applications. In *Proceedings of the 18th ACM symposium on Virtual reality software and technology*, pages 25–32, 2012.

[24] M. Kowalski, J. Naruniec, and M. Daniluk. Livescan3d: A fast and inexpensive 3d data acquisition system for multiple kinect v2 sensors. In *International Conference on 3D Vision*, pages 318–325, Oct 2015.

[25] M. Lindner, I. Schiller, A. Kolb, and R. Koch. Time-of-flight sensor calibration for accurate range sensing. *Computer Vision and Image Understanding*, 114(12):1318 – 1328, 2010.

[26] D. G. Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[27] A. Maimone and H. Fuchs. Encumbrance-free telepresence system with real-time 3d capture and display using commodity depth cameras. In *10th IEEE International Symposium on Mixed and Augmented Reality*, pages 137–146, Oct 2011.

[28] S. Miller, A. Teichman, and S. Thrun. Unsupervised extrinsic calibration of depth sensors in dynamic scenes. In *IEEE International Conference on Intelligent Robots and Systems*, pages 2695–2702, Nov 2013.

[29] R. A. Newcombe, D. Fox, and S. M. Seitz. Dynamicfusion: Reconstruction and tracking of non-rigid scenes in real-time. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 343–352, 2015.

[30] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohi, J. Shotton, S. Hodges, and A. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *10th IEEE International Symposium on Mixed and Augmented Reality*, pages 127–136, Oct 2011.

[31] S. Ostadabbas, C. Bulach, D. N. Ku, L. J. Anderson, and M. Ghovanloo. A passive quantitative measurement of air-

way resistance using depth data. In *IEEE-EMBS*, pages 5743–5747, 2014.

[32] S. Ostadabbas, N. Sebkhi, M. Zhang, S. Rahim, L. Anderson, F.-H. Lee, and M. Ghovanloo. A vision-based respiration monitoring system for passive airway resistance estimation. *IEEE Transactions on Biomedical Engineering*, PP(99):1–1, 2015.

[33] C. Pheatt. Intel® threading building blocks. *Journal of Computing Sciences in Colleges*, 23(4):298–298, 2008.

[34] F. Pomerleau, F. Colas, R. Siegwart, and S. Magnenat. Comparing icp variants on real-world data sets. *Autonomous Robots*, 34(3):133–148, 2013.

[35] A. Rafighi, S. Seifi, and O. Meruvia-Pastor. Automatic and adaptable registration of live rgbd video streams. In *Proceedings of the 8th ACM SIGGRAPH Conference on Motion in Games*, pages 243–250, 2015.

[36] E. Rublee, V. Rabaud, K. Konolige, and G. Bradski. Orb: An efficient alternative to sift or surf. In *International Conference on Computer Vision*, pages 2564–2571, Nov 2011.

[37] S. Rusinkiewicz, O. Hall-Holt, and M. Levoy. Real-time 3d model acquisition. *ACM Transactions on Graphics (TOG)*, 21(3):438–446, 2002.

[38] R. B. Rusu, N. Blodow, and M. Beetz. Fast point feature histograms (fpfh) for 3d registration. In *IEEE International Conference on Robotics and Automation*, pages 3212–3217, 2009.

[39] F. Sadlo, T. Weyrich, R. Peikert, and M. Gross. A practical structured light acquisition system for point-based geometry and texture. In *Proceedings Eurographics/IEEE VGTC Symposium Point-Based Graphics*, pages 89–145, 2005.

[40] W. J. Schroeder, K. Martin, and B. Lorensen. *The visualization toolkit : an object-oriented approach to 3D graphics*. Kitware, New York, 2006.

[41] S. Seifi, A. Rafighi, and O. Meruvia-Pastor. Derees: Real-time registration of rgbd images using image-based feature detection and robust 3d correspondence estimation and refinement. In *Proceedings of the 29th International Conference on Image and Vision Computing*, pages 136–141, 2014.

[42] S. M. Seitz, B. Curless, J. Diebel, D. Scharstein, and R. Szeliski. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 519–528, June 2006.

[43] M. Siudak and P. Rokita. A survey of passive 3d reconstruction methods on the basis of more than one image. *Machine Graphics and Vision*, 23:57–117, 2014.

[44] V. Soleimani, M. Mirmehdi, D. Damen, S. Hannuna, M. Camplani, J. Viner, and J. Dodd. Remote pulmonary function testing using a depth sensor. In *IEEE Conference on Biomedical Circuits and Systems*, pages 1–4, Oct 2015.

[45] P. H. S. Torr and A. Zisserman. MLESAC: A new robust estimator with application to estimating image geometry. *Computer Vision and Image Understanding*, 78:138–156, 2000.

[46] M. Weinmann, R. Ruiters, A. Osep, C. Schwartz, and R. Klein. Fusing structured light consistency and helmholtz normals for 3d reconstruction. In *Proceedings of the British Machine Vision Conference*, pages 1–12, 2012.

[47] M.-C. Yu, J.-L. Liou, S.-W. Kuo, M.-S. Lee, and Y.-P. Hung. Noncontact respiratory measurement of volume change using depth camera. In *IEEE-EMBS*, pages 2371–2374, 2012.

[48] Z. Zhang. Iterative point matching for registration of free-form curves and surfaces. *International journal of computer vision*, 13(2):119–152, oct 1994.

[49] M. Zollhöfer, M. Nießner, S. Izadi, C. Rehmann, C. Zach, M. Fisher, C. Wu, A. Fitzgibbon, C. Loop, C. Theobalt, et al. Real-time non-rigid reconstruction using an rgb-d camera. *ACM Transactions on Graphics (TOG)*, 33(4):156, 2014.